OIPE JAN 2 5 2008 PATENT & TRADEMARK OFFICE

TD
AF

S/N 10/687,291 **PATENT**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Michael J. Branson, et al.	Examiner: Haoshian Shih
Serial No.: 10/687,291	Group Art Unit: 2173
Filed: October 16, 2003	Confirmation Number: 8981
Title: Moving Data Between Views	Docket: ROC920030263US1

AMENDED APPEAL BRIEF
TO THE BOARD OF PATENT APPEALS AND INTERFERENCES
OF THE UNITED STATES PATENT AND TRADEMARK OFFICE

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This amended brief is presented in support of the Notice of Appeal filed on July 26, 2007, from the Final Rejection of claims 1-6 and 8-20 of the above-identified application, as set forth in the Final Office Action mailed on April 26, 2007, and in response to a Notification of Non-Compliant Appeal Brief mailed December 21, 2007.

The fee of $500.00 for filing an appeal brief has already been paid. Please charge any additional fees or credit overpayment to Deposit Account 09-0465. Appellant respectfully requests reversal of the Examiner's rejection of pending claims 1-6 and 8-20.

Table of Contents

1. Real Party in Interest

The real party in interest, in addition to the inventors, Michael J. Branson, George F. DeStefano, Gregory R. Hintermeister, and Andrew J. Streit, is the assignee, International Business Machines Corporation, a corporation organized and existing under and by virtue of the laws of the State of New York, and having an office and place of business at New Orchard Road, Armonk, New York 10504.

2. Related Appeals and Interferences

There are no other prior or pending appeals, interferences, or judicial proceedings, which may be related to, directly affect or be directly affected by, or have a bearing on the Board's decision.

3. Status of Claims

On July 26, 2007, Appellant appealed from the final rejection of claims 1-6 and 8-20 made in the Final Office Action dated April 26, 2007. Finally rejected claims 1-6 and 8-20 on appeal are set forth in the Claims Appendix. Claim 7 was canceled without prejudice or disclaimer.

4. Status of Amendments

Subsequent to the Final Office Action, on June 26, 2007, appellant filed an amendment under 37 CFR 1.116. In an Advisory Action dated July 10, 2007, the Supervisory Patent Examiner indicated that the amendment will not be entered.

5. Summary of Claimed Subject Matter

As described at page 2, last paragraph, lines 23-29 of appellant's specification, in Fig. 2, elements 205, 210, 215, and 220, in Fig. 4, elements 410, 415, 420, 422, 425, 430, 435, 440, and 445, a method, apparatus, system, and signal-bearing medium are provided that in an embodiment present a main view and at least one peek view of data. When the user selects to move the data from the main view to the peek view, a subset of data objects in the main view is selected based on an importance of the data objects and optionally based on the size of the peek view. The subset is copied to the peek view, and the peek view is dynamically updated when the data objects are updated.

With reference to claim 1, an embodiment of the invention comprises a method, which is described, for example, at page 2, third full paragraph, line 23 of the specification and in Fig. 4, elements 410, 415, 420, 422, 425, 430, 435, 440, and 445; wherein the method comprises:

selecting a subset of a first plurality of data objects based on a respective importance of each of the first plurality of respective data objects, wherein the first plurality of data objects are displayed in a main view, which is described, for example, at page 9, last full paragraph (lines 25-27) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, in Fig. 2, element 205, in Fig. 3, elements 130 and 326, and in Fig. 4, element 415;

copying the subset to a peek view, which is described, for example, at page 13, first full paragraph (lines 9-11) of the specification and in Fig. 4, element 420; and

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects, which is described, for example, at page 3, last full paragraph (lines 18-23), page 10, first full paragraph (lines 10-23) of the specification, page 14, first full paragraph (lines 7-12) of the specification, in Fig. 2, elements 205, 215, and 220, and in Fig. 4, elements 435 and 445.

With reference to claim 3, in an embodiment, the selecting further comprises: selecting the subset based on a size of the peek view, which is described, for example, at page

12, first full paragraph (lines 4-7) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, and in Fig. 4, element 415.

With reference to claim 6, an embodiment of the invention comprises an apparatus, which is described, for example, at page 2, last full paragraph (lines 23-24), at page 3, last partial paragraph (lines 24-27), at page 4, first partial paragraph, first full paragraph, second full paragraph, and last partial paragraph (lines 1-30), at page 5, first partial paragraph, first, second, and third full paragraphs (lines 1-29), at page 7, first full paragraph (lines 1-10), at page 8, first, second, third, and fourth full paragraphs (lines 8-28) of the specification and in Fig. 1, elements 100, 102, 115, 110, 126, 128, and 130, which recite:

"A method, apparatus, system, and signal-bearing medium are provided that in an embodiment present a main view and at least one peek view of data.

...

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a block diagram of an example system 100 for implementing an embodiment of the invention. The system 100 includes an electronic device 102 connected to a server 104 via a network 108. Although only one electronic device 102, one server 104, and one network 108 are shown, in other embodiments any number or combination of them may be present. Although the electronic device 102, the server 104, and the network 108 are illustrated in Fig. 1 as being discrete, separate components, in other embodiments some or all of their functions and elements may be combined.

In an embodiment, the electronic device 102 functions as a client that sends requests to and receives responses from the server 104. The electronic device 102 includes a processor 110, a storage device 115, an input device 120, and an output device 122, all connected directly or indirectly via a bus 125. The processor 110 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. The processor 110 executes instructions and includes that portion of the electronic device 102 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, the processor 110 typically includes a control unit that organizes data and program storage in

memory and transfers data and other information between the various parts of the electronic device 102. The processor 110 reads and/or writes code and data to/from the storage device 115, the network 108, the input device 120, and/or the output device 122.

Although the electronic device 102 is drawn to contain only a single processor 110 and a single bus 125, embodiments of the present invention apply equally to electronic devices that may have multiple processors and multiple buses with some or all performing different functions in different ways.

The storage device 115 represents one or more mechanisms for storing data. For example, the storage device 115 may include read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices, and/or other machine-readable media. In other embodiments, any appropriate type of storage device may be used. Although only one storage device 115 is shown, multiple storage devices and multiple types of storage devices may be present. Although the storage device 115 is shown in Fig. 1 as a single monolithic entity, the storage device 115 may in fact be distributed and/or hierarchical, as is known in the art. For example, the storage device 115 may exist in multiple levels of storage devices, and these levels of storage devices may be further divided by function, so that one level of storage device holds, e.g., instructions while another holds, e.g., non-instruction data which is used by the processor or processors. The storage device 115 may further be distributed and associated with different processors or sets of processors, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures. Further, although the electronic device 102 is drawn to contain the storage device 115, it may be distributed across other electronic devices, such as electronic devices connected to the network 108.

The storage device 115 includes a rendering agent 126, data 128, and tags 130, all of which may in various embodiments have any number of instances. Although the rendering agent 126, the data 128, and the tags 130 are all illustrated as being contained within the storage device 115 in the electronic device 102, in other embodiments some or all of them may be on different electronic devices and may be accessed remotely, e.g., via the network 108.

The rendering agent 126 receives data from the server 104, stores the received data in the data 128, and presents the data 128 in multiple views (as further described below with reference to Fig. 2) via the output device 122. The data 128 may be any appropriate data or information. The tags 130 specify how the data is to be presented in the multiple views. An example data structure for the tags 130 is further described below with reference to Fig. 3.

In an embodiment, the rendering agent 126 includes instructions capable of executing on the processor 110 or statements capable of being interpreted by instructions executing on the processor 110 to present the user interface as further described below with reference to Fig. 2, to manipulate the tags 130 as further described below with reference to Fig. 3, and to perform the functions as further described below with reference to Fig. 4. In another embodiment, the rendering agent 126 may be implemented in hardware via logic gates and/or other appropriate hardware techniques in lieu of or in addition to a processor-based system.

...

The electronic device 102 may be implemented using any suitable hardware and/or software, such as a personal computer. Portable computers, laptop or notebook computers, PDAs (Personal Digital Assistants), pocket computers, telephones, pagers, automobiles, teleconferencing systems, appliances, and mainframe computers are examples of other possible configurations of the electronic device 102. The hardware and software depicted in Fig. 1 may vary for specific applications and may include more or fewer elements than those depicted. For example, other peripheral devices such as audio adapters, or chip programming devices, such as EPROM (Erasable Programmable Read-Only Memory) programming devices may be used in addition to or in place of the hardware already depicted.

...

The various software components illustrated in Fig. 1 and implementing various embodiments of the invention may be implemented in a number of manners, including using various computer software applications, routines, components, programs, objects, modules, data structures, etc., referred to hereinafter as "computer programs," or simply "programs." The computer programs typically comprise one or more instructions that are resident at various times in various memory and storage devices in the electronic device 102, and that, when read and executed by one or more processors in the electronic device 102, cause the

electronic device to perform the steps necessary to execute steps or elements embodying the various aspects of an embodiment of the invention.

Moreover, while embodiments of the invention have and hereinafter will be described in the context of fully functioning electronic devices, the various embodiments of the invention are capable of being distributed as a program product in a variety of forms, and the invention applies equally regardless of the particular type of signal-bearing medium used to actually carry out the distribution. The programs defining the functions of this embodiment may be delivered to the electronic device 102 via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on a non-rewriteable storage medium, e.g., a read-only memory device attached to or within an electronic device, such as a CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on a rewriteable storage medium, e.g., a hard disk drive or diskette."

"electronic device 102,"

"storage device 115,"

"processor 110,"

"rendering agent 126,"

"data 128," and

"tags 130."

The apparatus of claim 6 comprises:

means for receiving a first plurality of data objects and a plurality of respective importance tags, which is described, for example, at page 3, last full paragraph (lines 13-23, at page 3, last partial paragraph (lines 24-27), at page 4, first partial paragraph, first full paragraph, second full paragraph, and last partial paragraph (lines 1-30), at page 5, first partial paragraph, first, second, and third full paragraphs (lines 1-29), at page 7, first full paragraph (lines 1-10), at page 8, first, second, third, and fourth full paragraphs (lines 8-28), at page 12, first full paragraph, lines 1-14, at page 14, first partial paragraph (lines 1-6) of the

specification, at Fig. 1, elements 100, 102, 115, 110, 126, 128, and 130, at Fig. 2, element 205, at Fig. 3, element 130, and at Fig. 4, elements 405, 435, 440, and 445, which recite:

"A rendering agent presents a main view of data objects and at least one peek view. In response to a pull command from the peek view, the rendering agent selects a subset of the data objects in the main view and copies the subset to the peek view. The rendering agent may select the subset based on an importance ranking of the data objects and optionally based on the size of the peek view that was the source of the pull command, with larger peek views receiving a larger subset. The rendering agent updates the subset in the peek view as the rendering agent receives updates to the data objects. The peek view may also issue a push command, which the rendering agent responds to by restoring the data objects to the main view. In this way, the user can monitor a subset of important information regarding dynamically changing data objects in the smaller peek view while continuing to work with other data objects in the larger main view.

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a block diagram of an example system 100 for implementing an embodiment of the invention. The system 100 includes an electronic device 102 connected to a server 104 via a network 108. Although only one electronic device 102, one server 104, and one network 108 are shown, in other embodiments any number or combination of them may be present. Although the electronic device 102, the server 104, and the network 108 are illustrated in Fig. 1 as being discrete, separate components, in other embodiments some or all of their functions and elements may be combined.

In an embodiment, the electronic device 102 functions as a client that sends requests to and receives responses from the server 104. The electronic device 102 includes a processor 110, a storage device 115, an input device 120, and an output device 122, all connected directly or indirectly via a bus 125. The processor 110 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. The processor 110 executes instructions and includes that portion of the electronic device 102 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, the

processor 110 typically includes a control unit that organizes data and program storage in memory and transfers data and other information between the various parts of the electronic device 102. The processor 110 reads and/or writes code and data to/from the storage device 115, the network 108, the input device 120, and/or the output device 122.

Although the electronic device 102 is drawn to contain only a single processor 110 and a single bus 125, embodiments of the present invention apply equally to electronic devices that may have multiple processors and multiple buses with some or all performing different functions in different ways.

The storage device 115 represents one or more mechanisms for storing data. For example, the storage device 115 may include read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices, and/or other machine-readable media. In other embodiments, any appropriate type of storage device may be used. Although only one storage device 115 is shown, multiple storage devices and multiple types of storage devices may be present. Although the storage device 115 is shown in Fig. 1 as a single monolithic entity, the storage device 115 may in fact be distributed and/or hierarchical, as is known in the art. For example, the storage device 115 may exist in multiple levels of storage devices, and these levels of storage devices may be further divided by function, so that one level of storage device holds, e.g., instructions while another holds, e.g., non-instruction data which is used by the processor or processors. The storage device 115 may further be distributed and associated with different processors or sets of processors, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures. Further, although the electronic device 102 is drawn to contain the storage device 115, it may be distributed across other electronic devices, such as electronic devices connected to the network 108.

The storage device 115 includes a rendering agent 126, data 128, and tags 130, all of which may in various embodiments have any number of instances. Although the rendering agent 126, the data 128, and the tags 130 are all illustrated as being contained within the storage device 115 in the electronic device 102, in other embodiments some or all of them may be on different electronic devices and may be accessed remotely, e.g., via the network 108.

The rendering agent 126 receives data from the server 104, stores the received data in the data 128, and presents the data 128 in multiple views (as further described below with reference to Fig. 2) via the output device 122. The data 128 may be any appropriate data or information. The tags 130 specify how the data is to be presented in the multiple views. An example data structure for the tags 130 is further described below with reference to Fig. 3.

In an embodiment, the rendering agent 126 includes instructions capable of executing on the processor 110 or statements capable of being interpreted by instructions executing on the processor 110 to present the user interface as further described below with reference to Fig. 2, to manipulate the tags 130 as further described below with reference to Fig. 3, and to perform the functions as further described below with reference to Fig. 4. In another embodiment, the rendering agent 126 may be implemented in hardware via logic gates and/or other appropriate hardware techniques in lieu of or in addition to a processor-based system.

. . .

The electronic device 102 may be implemented using any suitable hardware and/or software, such as a personal computer. Portable computers, laptop or notebook computers, PDAs (Personal Digital Assistants), pocket computers, telephones, pagers, automobiles, teleconferencing systems, appliances, and mainframe computers are examples of other possible configurations of the electronic device 102. The hardware and software depicted in Fig. 1 may vary for specific applications and may include more or fewer elements than those depicted. For example, other peripheral devices such as audio adapters, or chip programming devices, such as EPROM (Erasable Programmable Read-Only Memory) programming devices may be used in addition to or in place of the hardware already depicted.

. . .

The various software components illustrated in Fig. 1 and implementing various embodiments of the invention may be implemented in a number of manners, including using various computer software applications, routines, components, programs, objects, modules, data structures, etc., referred to hereinafter as "computer programs," or simply "programs." The computer programs typically comprise one or more instructions that are resident at various times in various memory and storage devices in the electronic device 102, and that, when read and executed by one or more processors in the electronic device 102, cause the

electronic device to perform the steps necessary to execute steps or elements embodying the various aspects of an embodiment of the invention.

Moreover, while embodiments of the invention have and hereinafter will be described in the context of fully functioning electronic devices, the various embodiments of the invention are capable of being distributed as a program product in a variety of forms, and the invention applies equally regardless of the particular type of signal-bearing medium used to actually carry out the distribution. The programs defining the functions of this embodiment may be delivered to the electronic device 102 via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on a non-rewriteable storage medium, e.g., a read-only memory device attached to or within an electronic device, such as a CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on a rewriteable storage medium, e.g., a hard disk drive or diskette.

...

The importance field 326 indicates the importance or ranking of the data object in the respective data object field 322. The rendering agent 126 uses the importance 326 to determine whether to include the associated data object in the subset of the data objects that are displayed in the peek view. In an embodiment, the rendering agent 126 uses the importance 326 along with the size of the peek view in selecting the subset. For example, the larger the size of the peek view, the larger the subset of selected data objects, where the rendering agent 126 selects those data objects with the highest importance first. In another embodiment, the importance 326 may be a binary decision, such as "yes" for displaying the data object in any peek view and "no" for not displaying the data object in any peek view. In the example of Fig. 2, only the task data object 242 and the status data object 244 are illustrated in the peek view 210 because only their associated records 305 and 310 have the highest importance. In an embodiment, the rendering agent 126 obtains the values for the importance 326 from the data source 160, but in other embodiments the values may be obtained from the user or from any other appropriate source.

...

agent 126 determines whether the event previously received at block 405 is a receive data from the data source 160 event. If the determination at block 435 is true, then the rendering agent 126 receives data from the data source 160 at the server 104 into the data 128 and continues to block 440 where the rendering agent 126 updates the tags 130 with the received data including updating the importance field 326 and the sort rule field 328 in an embodiment where they are included with the received data."

"electronic device" "102,"

"storage device" "115,"

"processor" "110,"

"rendering agent" "126,"

"data" "128,"

"tags" "130,"

"main view" "205,"

"receive event" "405"

"receive data from data source?" "435"

"update tags" "440," and

"copy data to main view or peek view depending on tags" "445."

The apparatus of claim 6 further comprises:

means for selecting a subset of the first plurality of data objects based on the importance tags and based on a peek view associated with a pull command, which is described, for example, at page 3, last full paragraph (lines 13-23), at page 3, last partial paragraph (lines 24-27), at page 4, first partial paragraph, first full paragraph, second full paragraph, and last partial paragraph (lines 1-30), at page 5, first partial paragraph, first, second, and third full paragraphs (lines 1-29), at page 7, first full paragraph (lines 1-10), at page 8, first, second, third, and fourth full paragraphs (lines 8-28), at page 10, first full paragraph, (lines 10-23), at page 13, first full paragraph (lines 4-9) of the specification, in

Fig. 1, elements 100, 102, 115, 110, 126, 128, and 130, in Fig. 2, elements 205, 210, 215, and 260, in Fig. 3, element 130, and in Fig. 4, elements 410 and 415, which recite:

"A rendering agent presents a main view of data objects and at least one peek view. In response to a pull command from the peek view, the rendering agent selects a subset of the data objects in the main view and copies the subset to the peek view. The rendering agent may select the subset based on an importance ranking of the data objects and optionally based on the size of the peek view that was the source of the pull command, with larger peek views receiving a larger subset. The rendering agent updates the subset in the peek view as the rendering agent receives updates to the data objects. The peek view may also issue a push command, which the rendering agent responds to by restoring the data objects to the main view. In this way, the user can monitor a subset of important information regarding dynamically changing data objects in the smaller peek view while continuing to work with other data objects in the larger main view.

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a block diagram of an example system 100 for implementing an embodiment of the invention. The system 100 includes an electronic device 102 connected to a server 104 via a network 108. Although only one electronic device 102, one server 104, and one network 108 are shown, in other embodiments any number or combination of them may be present. Although the electronic device 102, the server 104, and the network 108 are illustrated in Fig. 1 as being discrete, separate components, in other embodiments some or all of their functions and elements may be combined.

In an embodiment, the electronic device 102 functions as a client that sends requests to and receives responses from the server 104. The electronic device 102 includes a processor 110, a storage device 115, an input device 120, and an output device 122, all connected directly or indirectly via a bus 125. The processor 110 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. The processor 110 executes instructions and includes that portion of the electronic device 102 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, the

processor 110 typically includes a control unit that organizes data and program storage in memory and transfers data and other information between the various parts of the electronic device 102. The processor 110 reads and/or writes code and data to/from the storage device 115, the network 108, the input device 120, and/or the output device 122.

Although the electronic device 102 is drawn to contain only a single processor 110 and a single bus 125, embodiments of the present invention apply equally to electronic devices that may have multiple processors and multiple buses with some or all performing different functions in different ways.

The storage device 115 represents one or more mechanisms for storing data. For example, the storage device 115 may include read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices, and/or other machine-readable media. In other embodiments, any appropriate type of storage device may be used. Although only one storage device 115 is shown, multiple storage devices and multiple types of storage devices may be present. Although the storage device 115 is shown in Fig. 1 as a single monolithic entity, the storage device 115 may in fact be distributed and/or hierarchical, as is known in the art. For example, the storage device 115 may exist in multiple levels of storage devices, and these levels of storage devices may be further divided by function, so that one level of storage device holds, e.g., instructions while another holds, e.g., non-instruction data which is used by the processor or processors. The storage device 115 may further be distributed and associated with different processors or sets of processors, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures. Further, although the electronic device 102 is drawn to contain the storage device 115, it may be distributed across other electronic devices, such as electronic devices connected to the network 108.

The storage device 115 includes a rendering agent 126, data 128, and tags 130, all of which may in various embodiments have any number of instances. Although the rendering agent 126, the data 128, and the tags 130 are all illustrated as being contained within the storage device 115 in the electronic device 102, in other embodiments some or all of them may be on different electronic devices and may be accessed remotely, e.g., via the network 108.

The rendering agent 126 receives data from the server 104, stores the received data in the data 128, and presents the data 128 in multiple views (as further described below with reference to Fig. 2) via the output device 122. The data 128 may be any appropriate data or information. The tags 130 specify how the data is to be presented in the multiple views. An example data structure for the tags 130 is further described below with reference to Fig. 3.

In an embodiment, the rendering agent 126 includes instructions capable of executing on the processor 110 or statements capable of being interpreted by instructions executing on the processor 110 to present the user interface as further described below with reference to Fig. 2, to manipulate the tags 130 as further described below with reference to Fig. 3, and to perform the functions as further described below with reference to Fig. 4. In another embodiment, the rendering agent 126 may be implemented in hardware via logic gates and/or other appropriate hardware techniques in lieu of or in addition to a processor-based system.

...

Each of the peek views 210, 215, and 220 include a pull icon 260. When the pull icon 260 is selected, the rendering agent 126 selects a subset of the data objects (e.g., the data objects 240, 242, 244, 246, and 248) in the main view 205 and copies the subset to the peek view from which the pull icon 260 was selected. The rendering agent 126 then dynamically updates the subset as the data objects change. In the example illustrated in Fig. 2, the pull icon 260 in the peek view 210 was most recently selected, which resulted in the rendering agent 126 selecting the subset as the task data object 242 and the status data object 244 and copying that subset's data to the peek view 210. Notice that the data for the select data object 240, the user data object 246, and the type data object 248 are not part of the subset that the rendering agent 126 selected in this example. The rendering agent 126 makes the selection of the subset based on the tags 130 (Fig. 1). The pull icons in the peek views 215 and 220 were selected at some point in the past at a time when the main view 205 contained different data ("processes" data for the peek view 215 and "systems & group" data for the peek view 220).

...

If the determination at block 410 is true, then control continues to block 415 where the rendering agent 126 selects a subset of the data objects based on the associated importance tags 326 and optionally based on the peek view from which the pull event was

received. For example, in an embodiment, the rendering agent 126 uses the size of the associated peek view to determine how deep in the importance ranking to travel when selecting the subset, with larger peek views displaying more data objects. Control then continues to block 420 where the rendering agent 126 copies the selected subset of data objects to the peek view from which the pull event was received. The rendering agent 126 uses the sort rule field 328 in the tags 130 to sort the data in the displayed data objects. Control then continues to block 422 where the rendering agent 126 updates the view field 324 in the tags 130 to indicate the view(s) where the data is displayed. Control then returns to block 405 where the rendering agent 126 receives the next event, as previously described above."

"electronic device" "102,"

"storage device" "115,"

"processor" "110,"

"rendering agent" "126,"

"data" "128,"

"tags" "130,"

"pull event?" "410," and

"select subset of data using tags" "415."

The apparatus of claim 6 further comprises:

means for copying the subset from a main view to the peek view, which is described, for example, at page 3, last full paragraph (lines 13-23), at page 3, last partial paragraph (lines 24-27), at page 4, first partial paragraph, first full paragraph, second full paragraph, and last partial paragraph (lines 1-30), at page 5, first partial paragraph, first, second, and third full paragraphs (lines 1-29), at page 7, first full paragraph (lines 1-10), at page 8, first, second, third, and fourth full paragraphs (lines 8-28), at page 13, first full paragraph (lines 4-16) of the specification, in Fig. 1, elements 100, 102, 115, 110, 126, 128, and 130, and in Fig. 4, element 420, which recite:

"A rendering agent presents a main view of data objects and at least one peek view. In response to a pull command from the peek view, the rendering agent selects a subset of the data objects in the main view and copies the subset to the peek view. The rendering agent may select the subset based on an importance ranking of the data objects and optionally based on the size of the peek view that was the source of the pull command, with larger peek views receiving a larger subset. The rendering agent updates the subset in the peek view as the rendering agent receives updates to the data objects. The peek view may also issue a push command, which the rendering agent responds to by restoring the data objects to the main view. In this way, the user can monitor a subset of important information regarding dynamically changing data objects in the smaller peek view while continuing to work with other data objects in the larger main view.

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a block diagram of an example system 100 for implementing an embodiment of the invention. The system 100 includes an electronic device 102 connected to a server 104 via a network 108. Although only one electronic device 102, one server 104, and one network 108 are shown, in other embodiments any number or combination of them may be present. Although the electronic device 102, the server 104, and the network 108 are illustrated in Fig. 1 as being discrete, separate components, in other embodiments some or all of their functions and elements may be combined.

In an embodiment, the electronic device 102 functions as a client that sends requests to and receives responses from the server 104. The electronic device 102 includes a processor 110, a storage device 115, an input device 120, and an output device 122, all connected directly or indirectly via a bus 125. The processor 110 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. The processor 110 executes instructions and includes that portion of the electronic device 102 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, the processor 110 typically includes a control unit that organizes data and program storage in memory and transfers data and other information between the various parts of the electronic

device 102. The processor 110 reads and/or writes code and data to/from the storage device 115, the network 108, the input device 120, and/or the output device 122.

Although the electronic device 102 is drawn to contain only a single processor 110 and a single bus 125, embodiments of the present invention apply equally to electronic devices that may have multiple processors and multiple buses with some or all performing different functions in different ways.

The storage device 115 represents one or more mechanisms for storing data. For example, the storage device 115 may include read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices, and/or other machine-readable media. In other embodiments, any appropriate type of storage device may be used. Although only one storage device 115 is shown, multiple storage devices and multiple types of storage devices may be present. Although the storage device 115 is shown in Fig. 1 as a single monolithic entity, the storage device 115 may in fact be distributed and/or hierarchical, as is known in the art. For example, the storage device 115 may exist in multiple levels of storage devices, and these levels of storage devices may be further divided by function, so that one level of storage device holds, e.g., instructions while another holds, e.g., non-instruction data which is used by the processor or processors. The storage device 115 may further be distributed and associated with different processors or sets of processors, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures. Further, although the electronic device 102 is drawn to contain the storage device 115, it may be distributed across other electronic devices, such as electronic devices connected to the network 108.

The storage device 115 includes a rendering agent 126, data 128, and tags 130, all of which may in various embodiments have any number of instances. Although the rendering agent 126, the data 128, and the tags 130 are all illustrated as being contained within the storage device 115 in the electronic device 102, in other embodiments some or all of them may be on different electronic devices and may be accessed remotely, e.g., via the network 108.

The rendering agent 126 receives data from the server 104, stores the received data in the data 128, and presents the data 128 in multiple views (as further described below with

reference to Fig. 2) via the output device 122. The data 128 may be any appropriate data or information. The tags 130 specify how the data is to be presented in the multiple views. An example data structure for the tags 130 is further described below with reference to Fig. 3.

In an embodiment, the rendering agent 126 includes instructions capable of executing on the processor 110 or statements capable of being interpreted by instructions executing on the processor 110 to present the user interface as further described below with reference to Fig. 2, to manipulate the tags 130 as further described below with reference to Fig. 3, and to perform the functions as further described below with reference to Fig. 4. In another embodiment, the rendering agent 126 may be implemented in hardware via logic gates and/or other appropriate hardware techniques in lieu of or in addition to a processor-based system.

...

The electronic device 102 may be implemented using any suitable hardware and/or software, such as a personal computer. Portable computers, laptop or notebook computers, PDAs (Personal Digital Assistants), pocket computers, telephones, pagers, automobiles, teleconferencing systems, appliances, and mainframe computers are examples of other possible configurations of the electronic device 102. The hardware and software depicted in Fig. 1 may vary for specific applications and may include more or fewer elements than those depicted. For example, other peripheral devices such as audio adapters, or chip programming devices, such as EPROM (Erasable Programmable Read-Only Memory) programming devices may be used in addition to or in place of the hardware already depicted.

...

The various software components illustrated in Fig. 1 and implementing various embodiments of the invention may be implemented in a number of manners, including using various computer software applications, routines, components, programs, objects, modules, data structures, etc., referred to hereinafter as "computer programs," or simply "programs." The computer programs typically comprise one or more instructions that are resident at various times in various memory and storage devices in the electronic device 102, and that, when read and executed by one or more processors in the electronic device 102, cause the electronic device to perform the steps necessary to execute steps or elements embodying the various aspects of an embodiment of the invention.

Moreover, while embodiments of the invention have and hereinafter will be described in the context of fully functioning electronic devices, the various embodiments of the invention are capable of being distributed as a program product in a variety of forms, and the invention applies equally regardless of the particular type of signal-bearing medium used to actually carry out the distribution. The programs defining the functions of this embodiment may be delivered to the electronic device 102 via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on a non-rewriteable storage medium, e.g., a read-only memory device attached to or within an electronic device, such as a CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on a rewriteable storage medium, e.g., a hard disk drive or diskette.

...

If the determination at block 410 is true, then control continues to block 415 where the rendering agent 126 selects a subset of the data objects based on the associated importance tags 326 and optionally based on the peek view from which the pull event was received. For example, in an embodiment, the rendering agent 126 uses the size of the associated peek view to determine how deep in the importance ranking to travel when selecting the subset, with larger peek views displaying more data objects. Control then continues to block 420 where the rendering agent 126 copies the selected subset of data objects to the peek view from which the pull event was received. The rendering agent 126 uses the sort rule field 328 in the tags 130 to sort the data in the displayed data objects. Control then continues to block 422 where the rendering agent 126 updates the view field 324 in the tags 130 to indicate the view(s) where the data is displayed. Control then returns to block 405 where the rendering agent 126 receives the next event, as previously described above."

"electronic device" "102,"

"storage device" "115,"

"processor" "110,"

"rendering agent" "126,"

"data" "128,"

"tags" "130," and

"copy subset to peek view" "420."

The apparatus of claim 6 further comprises:

means for replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects, which is described, for example, at page 3, last full paragraph (lines 13-23), at page 3, last partial paragraph (lines 24-27), at page 4, first partial paragraph, first full paragraph, second full paragraph, and last partial paragraph (lines 1-30), at page 5, first partial paragraph, first, second, and third full paragraphs (lines 1-29), at page 7, first full paragraph (lines 1-10), at page 8, first, second, third, and fourth full paragraphs (lines 8-28), at page 10, first full paragraph (lines 10-23), at page 13, last partial paragraph (lines 29-30), at page 14, first partial paragraph (lines 1-6), at page 14, first full paragraph (lines 7-12) of the specification, in Fig. 1, elements 100, 102, 115, 110, 126, 128, and 130, in Fig. 2, elements 205, 215, and 220, and in Fig. 4, elements 435 and 445, which recite:

"A rendering agent presents a main view of data objects and at least one peek view. In response to a pull command from the peek view, the rendering agent selects a subset of the data objects in the main view and copies the subset to the peek view. The rendering agent may select the subset based on an importance ranking of the data objects and optionally based on the size of the peek view that was the source of the pull command, with larger peek views receiving a larger subset. The rendering agent updates the subset in the peek view as the rendering agent receives updates to the data objects. The peek view may also issue a push command, which the rendering agent responds to by restoring the data objects to the main view. In this way, the user can monitor a subset of important information regarding dynamically changing data objects in the smaller peek view while continuing to work with other data objects in the larger main view.

Referring to the Drawing, wherein like numbers denote like parts throughout the several views, Fig. 1 depicts a block diagram of an example system 100 for implementing an

embodiment of the invention. The system 100 includes an electronic device 102 connected to a server 104 via a network 108. Although only one electronic device 102, one server 104, and one network 108 are shown, in other embodiments any number or combination of them may be present. Although the electronic device 102, the server 104, and the network 108 are illustrated in Fig. 1 as being discrete, separate components, in other embodiments some or all of their functions and elements may be combined.

In an embodiment, the electronic device 102 functions as a client that sends requests to and receives responses from the server 104. The electronic device 102 includes a processor 110, a storage device 115, an input device 120, and an output device 122, all connected directly or indirectly via a bus 125. The processor 110 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. The processor 110 executes instructions and includes that portion of the electronic device 102 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, the processor 110 typically includes a control unit that organizes data and program storage in memory and transfers data and other information between the various parts of the electronic device 102. The processor 110 reads and/or writes code and data to/from the storage device 115, the network 108, the input device 120, and/or the output device 122.

Although the electronic device 102 is drawn to contain only a single processor 110 and a single bus 125, embodiments of the present invention apply equally to electronic devices that may have multiple processors and multiple buses with some or all performing different functions in different ways.

The storage device 115 represents one or more mechanisms for storing data. For example, the storage device 115 may include read only memory (ROM), random access memory (RAM), magnetic disk storage media, optical storage media, flash memory devices, and/or other machine-readable media. In other embodiments, any appropriate type of storage device may be used. Although only one storage device 115 is shown, multiple storage devices and multiple types of storage devices may be present. Although the storage device 115 is shown in Fig. 1 as a single monolithic entity, the storage device 115 may in fact be

distributed and/or hierarchical, as is known in the art. For example, the storage device 115 may exist in multiple levels of storage devices, and these levels of storage devices may be further divided by function, so that one level of storage device holds, e.g., instructions while another holds, e.g., non-instruction data which is used by the processor or processors. The storage device 115 may further be distributed and associated with different processors or sets of processors, as is known in any of various so-called non-uniform memory access (NUMA) computer architectures. Further, although the electronic device 102 is drawn to contain the storage device 115, it may be distributed across other electronic devices, such as electronic devices connected to the network 108.

The storage device 115 includes a rendering agent 126, data 128, and tags 130, all of which may in various embodiments have any number of instances. Although the rendering agent 126, the data 128, and the tags 130 are all illustrated as being contained within the storage device 115 in the electronic device 102, in other embodiments some or all of them may be on different electronic devices and may be accessed remotely, e.g., via the network 108.

The rendering agent 126 receives data from the server 104, stores the received data in the data 128, and presents the data 128 in multiple views (as further described below with reference to Fig. 2) via the output device 122. The data 128 may be any appropriate data or information. The tags 130 specify how the data is to be presented in the multiple views. An example data structure for the tags 130 is further described below with reference to Fig. 3.

In an embodiment, the rendering agent 126 includes instructions capable of executing on the processor 110 or statements capable of being interpreted by instructions executing on the processor 110 to present the user interface as further described below with reference to Fig. 2, to manipulate the tags 130 as further described below with reference to Fig. 3, and to perform the functions as further described below with reference to Fig. 4. In another embodiment, the rendering agent 126 may be implemented in hardware via logic gates and/or other appropriate hardware techniques in lieu of or in addition to a processor-based system.

. . .

The electronic device 102 may be implemented using any suitable hardware and/or software, such as a personal computer. Portable computers, laptop or notebook computers,

PDAs (Personal Digital Assistants), pocket computers, telephones, pagers, automobiles, teleconferencing systems, appliances, and mainframe computers are examples of other possible configurations of the electronic device 102. The hardware and software depicted in Fig. 1 may vary for specific applications and may include more or fewer elements than those depicted. For example, other peripheral devices such as audio adapters, or chip programming devices, such as EPROM (Erasable Programmable Read-Only Memory) programming devices may be used in addition to or in place of the hardware already depicted.

...

The various software components illustrated in Fig. 1 and implementing various embodiments of the invention may be implemented in a number of manners, including using various computer software applications, routines, components, programs, objects, modules, data structures, etc., referred to hereinafter as "computer programs," or simply "programs." The computer programs typically comprise one or more instructions that are resident at various times in various memory and storage devices in the electronic device 102, and that, when read and executed by one or more processors in the electronic device 102, cause the electronic device to perform the steps necessary to execute steps or elements embodying the various aspects of an embodiment of the invention.

Moreover, while embodiments of the invention have and hereinafter will be described in the context of fully functioning electronic devices, the various embodiments of the invention are capable of being distributed as a program product in a variety of forms, and the invention applies equally regardless of the particular type of signal-bearing medium used to actually carry out the distribution. The programs defining the functions of this embodiment may be delivered to the electronic device 102 via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on a non-rewriteable storage medium, e.g., a read-only memory device attached to or within an electronic device, such as a CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on a rewriteable storage medium, e.g., a hard disk drive or diskette.

...

Each of the peek views 210, 215, and 220 include a pull icon 260. When the pull icon 260 is selected, the rendering agent 126 selects a subset of the data objects (e.g., the data objects 240, 242, 244, 246, and 248) in the main view 205 and copies the subset to the peek view from which the pull icon 260 was selected. The rendering agent 126 then dynamically updates the subset as the data objects change. In the example illustrated in Fig. 2, the pull icon 260 in the peek view 210 was most recently selected, which resulted in the rendering agent 126 selecting the subset as the task data object 242 and the status data object 244 and copying that subset's data to the peek view 210. Notice that the data for the select data object 240, the user data object 246, and the type data object 248 are not part of the subset that the rendering agent 126 selected in this example. The rendering agent 126 makes the selection of the subset based on the tags 130 (Fig. 1). The pull icons in the peek views 215 and 220 were selected at some point in the past at a time when the main view 205 contained different data ("processes" data for the peek view 215 and "systems & group" data for the peek view 220).

...

If the determination at block 425 is false, then the event previously received at block 405 is not a push event, so control continues to block 435 where the rendering agent 126 determines whether the event previously received at block 405 is a receive data from the data source 160 event. If the determination at block 435 is true, then the rendering agent 126 receives data from the data source 160 at the server 104 into the data 128 and continues to block 440 where the rendering agent 126 updates the tags 130 with the received data including updating the importance field 326 and the sort rule field 328 in an embodiment where they are included with the received data.

Control then continues to block 445 where the rendering agent 126 copies the data 128 to the main view 205 or the peek view (e.g., one the peek views 210, 215 or 220) depending on the contents of the view field 324 and the importance field 326 associated with the received data. The rendering agent 126 further sorts the data objects in the peek view using the sort rule 328. In this way, the data in the main view 205 and/or the peek views is dynamically updated. Control then returns to block 405 where the rendering agent 126 receives the next event, as previously described above."

"electronic device" "102,"

"storage device" "115,"

"processor" "110,"

"rendering agent" "126,"

"data" "128,"

"tags" "130,"

"main view" "205,

"peek view processes" "215,"

"peek view systems & groups" "system a OK" "group c OK" "220"

"receive data from data source?" "435," and

"copy data to main view or peek view depending on tags" "445."

With reference to claim 8, in an embodiment, the means for selecting based on the peek view is further based on a size of the peek view, which is described, for example, at page 12, first full paragraph (lines 4-7) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, and in Fig. 4, element 415.

With reference to claim 11, an embodiment of the invention comprises a storage medium encoded with instructions, which is described, for example, at page 5, third full paragraph, (lines 22-25) of the specification, at page 8, second and third full paragraphs and the final partial paragraph (lines 17-28) of the specification, an in Fig. 1, element 115.

With further reference to claim 11, the instructions when executed comprise:

selecting a subset of a first plurality of data objects in response to a pull command from a peek view, wherein the first plurality of data objects are displayed in a main view, which is described, for example, at page 12, last partial paragraph (lines 27-30) of the specification, at page 13, first partial paragraph and first full paragraph (lines 1-9) of the specification, in Fig. 2, element 205, 210, 215, 220, and 260, and in Fig. 4, elements 410 and 415;

copying the subset to a peek view, which is described, for example, at page 13, first full paragraph (lines 9-11) of the specification, in Fig. 2, elements 210, 215, and 220, and in Fig. 4, element 420; and

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects, which is described, for example, at page 3, last full paragraph (lines 18-23) of the specification, page 10, first full paragraph (lines 10-23) of the specification, page 14, first full paragraph (lines 7-12) of the specification, in Fig. 2, elements 205, 215, and 220, and in Fig. 4, elements 435 and 445.

With reference to claim 12, the selecting further comprises: selecting the subset based on a plurality of importance tags associated with the respective first plurality of respective data objects, wherein the respective importance tags specify a ranking of the first plurality of respective data objects, which is described, for example, at page 3, last full paragraph (lines 15-16) of the specification, at page 10, first full paragraph (lines 10-23) of the specification, at page 12, first full paragraph (lines 1-14) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, in Fig. 2, element 205, in Fig. 3, elements 130 and 326, and in Fig. 4, elements 410 and 415.

With reference to claim 13, the selecting further comprises: selecting the subset based on the plurality of importance tags and a size of the peek view, which is described, for example, at page 12, first full paragraph (lines 4-7) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, and in Fig. 4, element 415.

With reference to claim 16, an embodiment of the invention comprises an electronic device, which is described, for example, at page 3, last partial paragraph (lines 24-27) of the specification and in Fig. 1, element 102;

a processor, which is described, for example, at page 4, first and second full paragraphs (lines 6-22) of the specification and in Fig. 1, element 110; and

a storage device encoded with instructions, which is described, for example, at page 4, last partial paragraph (lines 23-30) of the specification, at page 5, first partial paragraph, first full paragraph, second full paragraph, and third full paragraph (lines 1-29) of the specification, and in Fig. 1, elements 115 and 126.

With further reference to claim 16, the instructions when executed on the processor comprise:

selecting a subset of a first plurality of data objects in response to a pull command from a peek view, wherein the first plurality of data objects are displayed in a main view, which is described, for example, at page 12, last partial paragraph (lines 27-30) of the specification, at page 13, first partial paragraph and first full paragraph (lines 1-9) of the specification, in Fig. 2, elements 205, 210, 215, 220, and 260, and in Fig. 4, elements 410 and 415;

copying the subset to a peek view, which is described, for example, at page 13, first full paragraph (lines 9-11) of the specification and in Fig. 4, element 420;

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects, which is described, for example, at page 3, last full paragraph (lines 18-23) of the specification, page 10, first full paragraph (lines 10-23) of the specification, page 14, first full paragraph (lines 7-12) of the specification, in Fig. 2, elements 205, 215, and 220, and in Fig. 4, element 445;

receiving an update to the first plurality of data objects, which is described, for example, at page 3, last full paragraph (lines 18-23), at page 14, first partial paragraph (lines 1-6) of the specification, at Fig. 1, element 128, at Fig. 2, element 205, and at Fig. 4, elements 405, 435, and 440; and

modifying the subset in the peek view based on the update, which is described, for example, at page 10, first full paragraph (lines 13-14) in the specification, at page 14, first full paragraph (lines 7-13) of the specification, in Fig. 2, elements 210, 215, and 220, and at Fig. 4, elements 405, 435, 440, and 445.

With reference to claim 17, the selecting further comprises: selecting the subset based on a plurality of importance tags associated with the respective first plurality of respective data objects, wherein the respective importance tags specify a ranking of the first plurality of respective data objects, which is described, for example, at page 3, last full paragraph (lines 15-16) of the specification, at page 10, first full paragraph (lines 10-23) of the specification, at page 12, first full paragraph (lines 1-14) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, in Fig. 2, element 205, in Fig. 3, elements 130 and 326, and in Fig. 4, elements 410 and 415.

With reference to claim 18, the selecting further comprises: selecting the subset based on the plurality of importance tags and a size of the peek view, which is described, for example, at page 12, first full paragraph (lines 4-7) of the specification, at page 13, first full paragraph (lines 4-9) of the specification, and in Fig. 4, element 415.

6. Grounds of Rejection to be Reviewed on Appeal

1. Whether claims 1-6 and 8-20 are unpatentable under 35 U.S.C. 102(e) as being anticipated by Gegner (WO 2003/104966 A3), hereinafter "Gegner."

7. Argument

A) The Applicable Law

Anticipation requires the disclosure in a single prior art reference of each element of the claim under consideration. *In re Dillon* 919 F.2d 688, 16 USPQ 2d 1897, 1908 (Fed. Cir. 1990) (en banc), cert. denied, 500 U.S. 904 (1991). It is not enough, however, that the prior art reference discloses all the claimed elements in isolation. Rather, "[a]nticipation requires the presence in a single prior reference disclosure of each and every element of the claimed invention, *arranged as in the claim.*" *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 730 F.2d 1452, 221 USPQ 481, 485 (Fed. Cir. 1984) (citing *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542, 220 USPQ 193 (Fed. Cir. 1983)) (emphasis added). "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989); MPEP § 2131.

The Examiner has the burden under 35 U.S.C. § 103 to establish a *prima facie* case of obviousness. *In re Fine*, 837 F.2d 1071, 1074, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988). To do that the Examiner must show that some objective teaching in the prior art or some knowledge generally available to one of ordinary skill in the art would lead an individual to combine the relevant teaching of the references. *Id.*

The *Fine* court stated that:

> Obviousness is tested by "what the combined teaching of the references would have suggested to those of ordinary skill in the art." *In re Keller*, 642 F.2d 413, 425, 208 USPQ 871, 878 (CCPA 1981). But it "cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination." *ACS Hosp. Sys.*, 732 F.2d at 1577, 221 USPQ at 933. And "teachings of references can be combined *only* if there is some suggestion or incentive to do so." *Id.* (emphasis in original).

The M.P.E.P. adopts this line of reasoning, stating that

> In order for the Examiner to establish a *prima facie* case of obviousness, three base criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation

> of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *M.P.E.P.* § 2142 (citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir. 1991)).

An invention can be obvious even though the suggestion to combine prior art teachings is not found in a specific reference. *In re Oetiker*, 24 USPQ2d 1443 (Fed. Cir. 1992). At the same time, however, although it is not necessary that the cited references or prior art specifically suggest making the combination, there must be some teaching somewhere which provides the suggestion or motivation to combine prior art teachings and applies that combination to solve the same or similar problem which the claimed invention addresses. One of ordinary skill in the art will be presumed to know of any such teaching. (See, e.g., *In re Nilssen*, 851 F.2d 1401, 1403, 7 USPQ2d 1500, 1502 (Fed. Cir. 1988) and *In re Wood*, 599 F.2d 1032, 1037, 202 USPQ 171, 174 (CCPA 1979)).

A factor cutting against a finding of motivation to combine or modify the prior art is when the prior art teaches away from the claimed combination. A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path the applicant took. *In re Gurley*, 27 F.3d 551, 31 USPQ 2d 1130, 1131 (Fed. Cir. 1994); *United States v. Adams*, 383 U.S. 39, 52, 148 USPQ 479, 484 (1966); *In re Sponnoble*, 405 F.2d 578, 587, 160 USPQ 237, 244 (C.C.P.A. 1969); *In re Caldwell*, 319 F.2d 254, 256, 138 USPQ 243, 245 (C.C.P.A. 1963).

The test for obviousness under § 103 must take into consideration the invention as a whole; that is, one must consider the particular problem solved by the combination of elements that define the invention. *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 USPQ 543, 551 (Fed. Cir. 1985). Furthermore, claims must be interpreted in light of the specification, claim language, other claims and prosecution history. *Panduit Corp. v. Dennison Mfg. Co.*, 810 F.2d 1561, 1568, 1 USPQ2d 1593, 1597 (Fed. Cir. 1987), *cert. denied*, 481 U.S. 1052 (1987). At the same time, a prior patent cited as a § 103 reference must be considered in its entirety, "*i.e.* as a *whole*, including portions that lead away from the invention." *Id.* That is, the Examiner must, as one of the inquiries pertinent to any

obviousness inquiry under 35 U.S.C. § 103, recognize and consider not only the similarities but also the critical differences between the claimed invention and the prior art. *In re Bond*, 910 F.2d 831, 834, 15 USPQ2d 1566, 1568 (Fed. Cir. 1990), *reh'g denied*, 1990 U.S. App. LEXIS 19971 (Fed. Cir. 1990). Finally, the Examiner must avoid hindsight. *Id.*

As explained in M.P.E.P. § 2112, the express, implicit, and inherent disclosures of a prior art reference may be relied upon in the rejection of claims under 35 U.S.C. 102 or 103. But, the fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic. In re Rijckaert, 9 F.3d 1531, 1534, 28 USPQ2d 1955, 1957 (Fed. Cir. 1993). Further, "[i]n relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art." Ex parte Levy, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990) (emphasis in original).

B) Discussion of the Rejections

1. Claims 1-6 and 8-20 are rejected under 35 U.S.C. 102(b) as being anticipated by Gegner (WO 2003/104,966 A3).

Claims 1-6 and 8-20

Appellant respectfully submits that claims 1-6 and 8-20 are patentable over Gegner because all of the elements of claims 1-6 and 8-20 are not taught or suggested by Gegner for the reasons argued below.

Claim 1 recites: "selecting a subset of a first plurality of data objects ...; copying the subset to a peek view; and replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects." Since claim 1 recites that the subset is copied from the first plurality of data objects and the second plurality of data objects are different from the first plurality of data objects, once the first plurality of data objects have been replaced, the subset in the peek view is different from the second plurality of data objects in the main view.

In contrast, in Gegner, the left-hand side of Fig. 5 (which the Examiner relies on for a peek view) is identical to a portion of the right-hand side of Fig. 5 (which the Office Action relies on for a main view). Thus, the right-hand side of Fig. 5 was not replaced by different data objects, so Gegner does not teach or suggest "replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects," as recited in claim 1.

Claims 6, 11, and 16 include similar elements as argued above for claim 1 and are patentable over the references for similar reasons. Claims 2-5, 8-10, 12-15, and 17-20 are dependent on claims 1, 6, 11, and 16, respectively, and are patentable for the reasons argued above, plus the elements in the claims.

<u>Claims 1-6, 8-10, 12, 13, 17, and 18</u>

Appellant respectfully submits that claims 1-6, 8-10, 12, 13, 17, and 18 are patentable over Gegner because all of the elements of claims 1-6, 8-10, 12, 13, 17, and 18 are not taught or suggested by Gegner for the reasons argued below.

Claim 1 recites "selecting a subset of a first plurality of data objects based on a respective importance of each of the first plurality of respective data objects, wherein the first plurality of data objects are displayed in a main view; copying the subset to a peek view; and replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects."

Thus, in claim 1, the data objects that are copied to the peek view are selected based on the importance of the objects, and the entire first plurality of data objects remain in the main view until all are replaced with the second plurality of data objects. Hence, data objects are not selectively suppressed; instead, data objects are selectively copied to an additional (peek) view.

In contrast, Gegner suppresses the display of less important objects, as recited in Gegner at page 3, lines 12-16: "the objects are arranged within a fixed hierarchy so that when the display resource on the display screen is not sufficient, objects can be automatically

suppressed, that is, starting with the lowest hierarchical level. As a result of the automatic suppression of less important objects, the display resource thus made available ensures a clearer display."

Hence, Gegner suppresses the display of objects based on importance while claim 1 copies objects to a peek view based on importance, so Gegner describes the opposite of claim 1, and thus Gegner teaches away from claim 1.

Claims 6, 12, and 17 include similar elements as argued above for claim 1 and are patentable over Gegner for similar reasons. Claims 2-5, 8-10, 13, and 18 are dependent on claims 1, 6, 12, and 17, respectively, and are patentable for the reasons argued above, plus the elements in the claims.

Claims 3, 8, 13, and 18

Appellant respectfully submits that claims 3, 8, 13, and 18 are patentable over Gegner because all of the elements of claims 3, 8, 13, and 18 are not taught or suggested by Gegner for the reasons argued below.

Claim 3 recites: "wherein the selecting further comprises: selecting the subset based on a size of the peek view." Claim 3 is dependent on claim 1, which recites: selecting a subset of a first plurality of data objects ... , wherein the first plurality of data objects are displayed in a main view; copying the subset to a peek view." Thus, in claim 3, the data objects that are copied to the peek view are selected based on the size of the peek view into which the subset is copied.

The Examiner relies on Gegner, page 3, lines 27-29, which recites: "when a group of objects is reduced in such a manner that readability of all objects of this group is no longer ensured, less important objects of the group are automatically eliminated, so as to ensure optimum readability of the remaining objects combined in the group." Thus, Gegner picks those objects to eliminate that are less important and Gegner performs this at a time when readability is no longer ensured. In contrast, claim 3 (and claim 1 on which claim 3 depends) does not eliminate objects. Instead, claim 3 selects a subset of objects based on the size of the peek view into which claim 1 copies the subset. Thus, the Gegner eliminating objects at a

time when readability is not ensured does not teach or suggest selecting objects to copy to a peek view based on the size of the peek view, as recited in claim 3 and in claim 1, on which claim 3 depends.

Claims 8, 13, and 18 recite similar elements as argued above for claim 3 and are patentable over Gegner for similar reasons.

Conclusion

Appellant respectfully requests reversal of the above rejections. If the Board is of the opinion that any rejected claim may be allowable in amended form, then appellant also respectfully requests a statement to that effect.

Respectfully submitted,

Michael J. Branson, et al.
By their Representative,

Date January 22, 2008 By [signature]

Owen J. Gamon
Reg. No.: 36,143
phone: 651-645-7135
fax: 651-457-5622

IBM Corporation
Intellectual Property Law
Dept. 917, Bldg. 006-1
3605 Highway 52 North
Rochester, MN 55901

Certificate under 37 C.F.R. 1.8: I hereby certify that this correspondence is being transmitted via facsimile (Fax no: 571-273-8300) to the Commissioner of Patents or is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner of Patents, Alexandria, VA 22313-1450 on January 22, 2008.

Name Owen J. Gamon Signature [signature]

8. CLAIMS APPENDIX

1. A method comprising:

selecting a subset of a first plurality of data objects based on a respective importance of each of the first plurality of respective data objects, wherein the first plurality of data objects are displayed in a main view;

copying the subset to a peek view; and

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects.

2. The method of claim 1, wherein the selecting is in response to a pull command at the peek view.

3. The method of claim 1, wherein the selecting further comprises:

selecting the subset based on a size of the peek view.

4. The method of claim 1, further comprising:

receiving an update to the plurality of data objects; and

modifying the subset in the peek view based on the update.

5. The method of claim 4, further comprising:

re-selecting the subset based on a change to the importance, wherein the receiving further receives the change to the importance.

6. An apparatus comprising:

means for receiving a first plurality of data objects and a plurality of respective importance tags;

means for selecting a subset of the first plurality of data objects based on the importance tags and based on a peek view associated with a pull command;

means for copying the subset from a main view to the peek view; and

means for replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects.

8. The apparatus of claim 6, wherein the means for selecting based on the peek view is further based on a size of the peek view.

9. The apparatus of claim 6, further comprising:

means for copying the subset from the peek view to the main view in response to a push command associated with the peek view.

10. The apparatus of claim 6, further comprising:

means for receiving an update to the first plurality of data objects; and

means for modifying the subset in the peek view based on the update.

11. A storage medium encoded with instructions, wherein the instructions when executed comprise:

selecting a subset of a first plurality of data objects in response to a pull command from a peek view, wherein the first plurality of data objects are displayed in a main view;

copying the subset to a peek view; and

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects.

12. The storage medium of claim 11, wherein the selecting further comprises:

selecting the subset based on a plurality of importance tags associated with the respective first plurality of respective data objects, wherein the respective importance tags specify a ranking of the first plurality of respective data objects.

13. The storage medium of claim 12, wherein the selecting further comprises:

selecting the subset based on the plurality of importance tags and a size of the peek view.

14. The storage medium of claim 11, further comprising:

receiving an update to the first plurality of data objects; and

modifying the subset in the peek view based on the update.

15. The storage medium of claim 14, further comprising:

modifying the first plurality of data objects in the main view based on the update.

16. An electronic device comprising:

a processor; and

a storage device encoded with instructions, wherein the instructions when executed on the processor comprise:

selecting a subset of a first plurality of data objects in response to a pull command from a peek view, wherein the first plurality of data objects are displayed in a main view,

copying the subset to a peek view,

replacing the first plurality of data objects in the main view with a second plurality of data objects, wherein the second plurality of data objects are different from the first plurality of data objects,

receiving an update to the first plurality of data objects, and

modifying the subset in the peek view based on the update.

17. The electronic device of claim 16, wherein the selecting further comprises:

selecting the subset based on a plurality of importance tags associated with the respective first plurality of respective data objects, wherein the respective importance tags specify a ranking of the first plurality of respective data objects.

18. The electronic device of claim 17, wherein the selecting further comprises:

selecting the subset based on the plurality of importance tags and a size of the peek view.

19. The electronic device of claim 16, wherein the instructions further comprise:

copying the subset back to the main view in response to a push command from the peek view.

20. The electronic device of claim 16, wherein the instructions further comprise:

sorting data in the subset in the peek view based on a sort rule associated with the data.

9. EVIDENCE APPENDIX

None.

10. RELATED PROCEEDINGS APPENDIX

None.